

Alpgen and D0.

Gérald Grenier
GDR-SUSY TOOLS meeting
CERN – March 2nd 2007

Introduction

Usually :

- generator's authors provide an executable (or source code with makefile) well suited for stand-alone running.
- Programs are in Fortran or eventually in C++.
- Experimental collaborations run generators on computer farms for automated mass production.
- The generators (or some portions of them) are embedded in a more global software (the experimental collaboration framework).
- The framework is in C++.

Alpgen : a quick technical look

Alpgen is a FOME + Parton Shower (matching) generator.

First step = produce FOME partonic events

Second step = execute Parton Shower and matching (Pythia or Herwig)

The alpgen package distribution comes with many directories

alplib/ the core software for FOME and Parton Shower matching

wjetlib/ wjetwork/ w+jets specifics ---> produces executable wjetgen (FOME specific)

zjetlib/ zjetwork/ z+jets specifics ---> produces executable zjetgen (FOME specific)

.....

pylib/ pythia+specific Parton Shower matching code

herlib/ herwig+specific Parton Shower matching code

wjetlib, zjetlib,... have different routines with names in common to implement the specific points of each processes.

In alpgen 2.1X, there are 16 different processes implemented
16 different executables.

Note to generator's authors : Alpgen structure cries for C++ features.

Alpgen in D0

A C++ executable reads input from a configuration file then :

- Chooses the right Alpgen executables.
- Runs the independant stand-alone Alpgen executable (eventually more than once).
- Runs the Parton-Shower matching
- Stores the kept event into a binary file using specific D0-I/O software.

Note :

FOME part of alpgen is the original Alpgen code.

Parton Shower is linked in the main executable : needs some rewriting of the code.

The C++ executable runs in batch on computer farms.

Wishes

Extract from Alpgen Parton Shower matching :

```
SUBROUTINE UPINIT
```

```
.....
```

```
WRITE(*,*) 'INPUT NAME OF FILE CONTAINING EVENTS'
```

```
WRITE(*,*) '(FOR "file.unw" ENTER "file")'
```

```
READ(*,*) FILENAME
```

```
.....
```

```
END
```

Not practical for running in computer center or on the grid.
Having a way to configure by software is better.

Extract from Herwig :

```
C-----  
      SUBROUTINE HWUGUP  
C-----  
C      Subroutine to handle termination of HERWIG if reaches end of event  
C      file  
C-----  
      INCLUDE 'HERWIG65.INC'  
C--reset the number of events to the correct value  
      NEVHEP = NEVHEP-1  
C--output information on the events  
      CALL HWEFIN  
C--run users end code  
      CALL HWAEND  
      STOP  
      END
```

Computer grid/centers will assume the program failed.
Not a good idea to normally end a program by crashing it.

An example

Look for Lepton Flavor Violating Higgs decay in $\tau\mu$ with the τ decaying in μ

Main background = $Z \rightarrow \mu\mu$

Select 2 isolated muons : one with $P_t > 20$ GeV, one with $P_t > 8$ GeV,
combine them with MET to reconstruct a Higgs candidate.

Note : work in progress

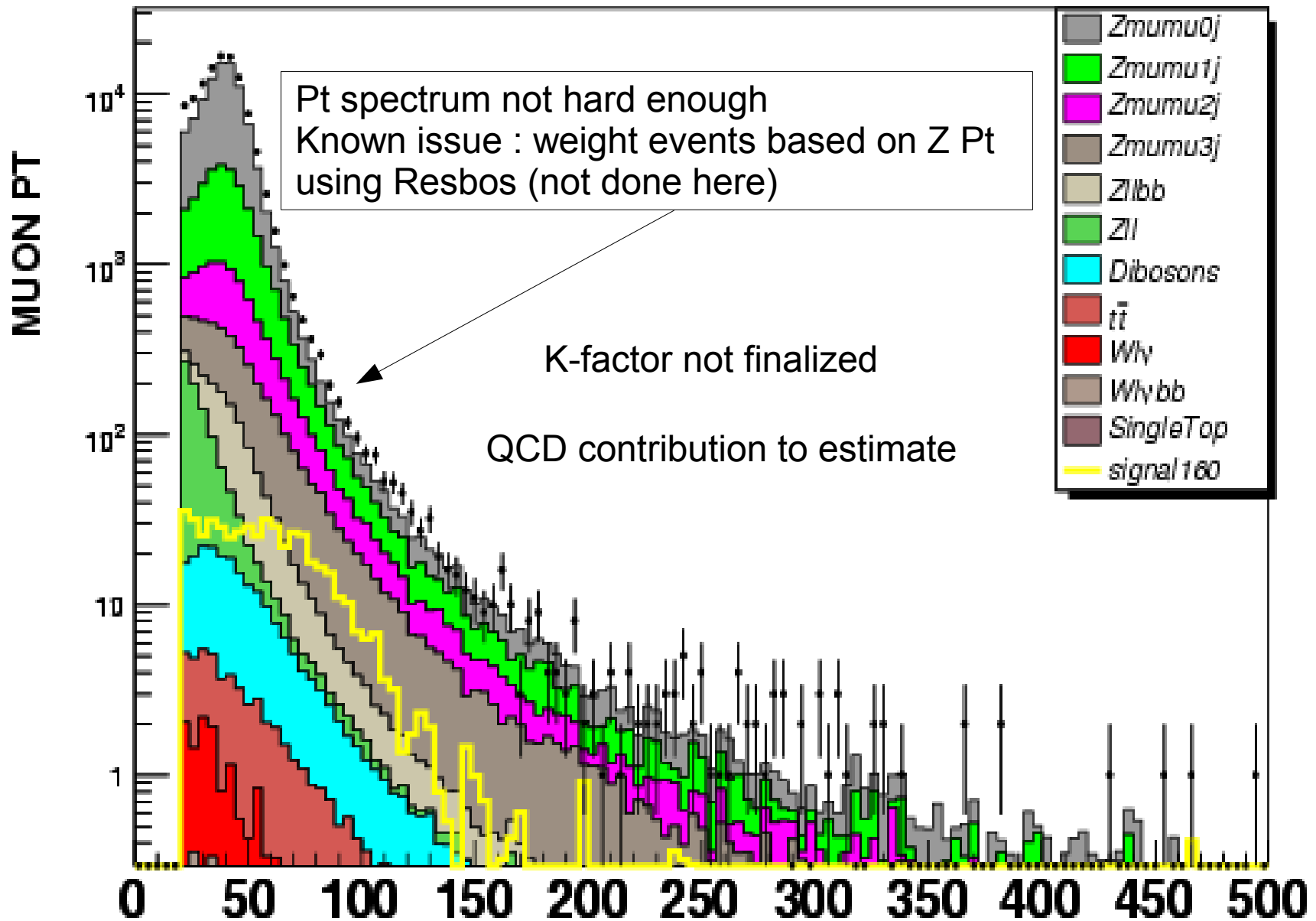
Signal	Xsec (pb)	Generator
$Z \rightarrow \ell\ell$ (15-60)	1168.1	Alpgen 0-3lp
$Z \rightarrow \ell\ell$ (60-130)	196.7	Alpgen 0-3lp
$Z \rightarrow \ell\ell$ (130-250)	4.2	Alpgen 0-3lp
$Z \rightarrow \ell\ell$ (250-1960)	0.4	Alpgen 0-3lp
$W \rightarrow \ell \nu$	5981.2	Alpgen 0-5lp
$Z \rightarrow \nu\nu$	1137.2	Alpgen 0-5lp
$t\bar{t}$	5.0	Alpgen 0-2lp
Diboson	16.6	Pythia
Single top	2.5	ConpHEP
$W \rightarrow \ell \nu$ + bb	31.6	Alpgen 0-3lp
$Z \rightarrow \ell\ell$ + 2b(60-130)	4.7	Alpgen 0-2lp
$Z \rightarrow \nu\nu$ + 2b	9.1	Alpgen 0-2lp

Missing $Z \rightarrow \tau\tau$ MC
for 250-1960 GeV
mass bin

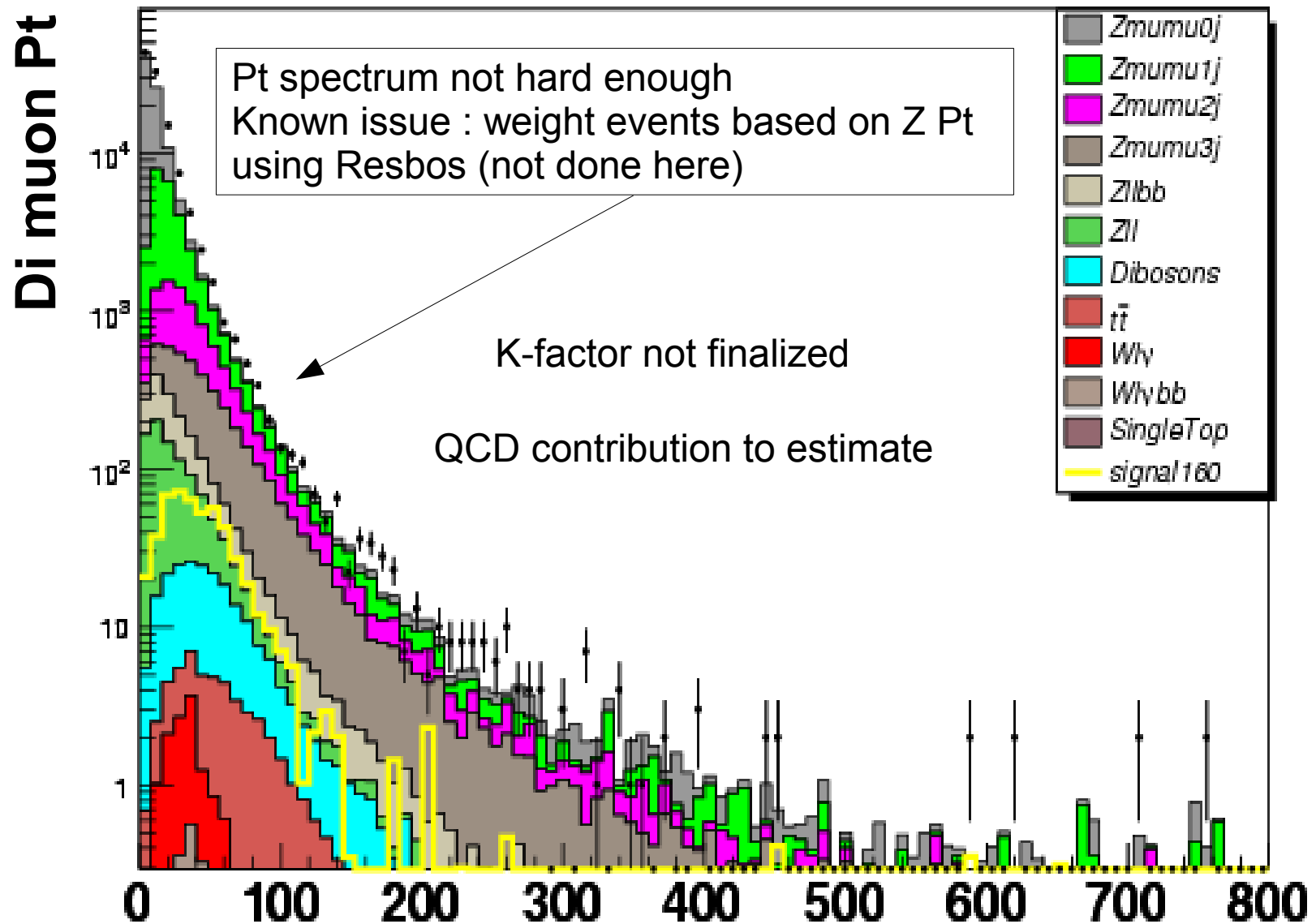
Alpgen 2.05 and 2.06

Apply K-factor of the order of 1.3

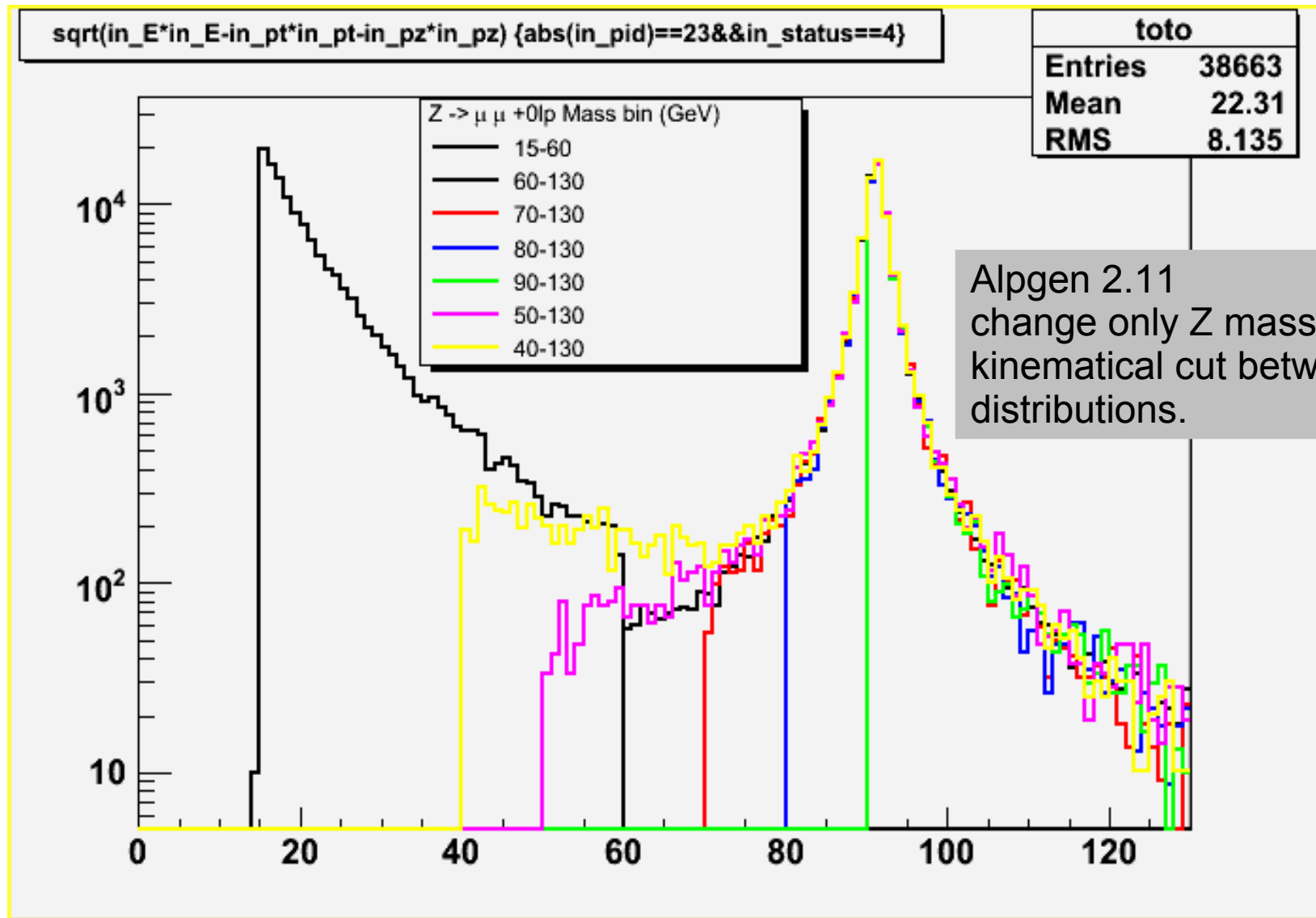
Muon Pt



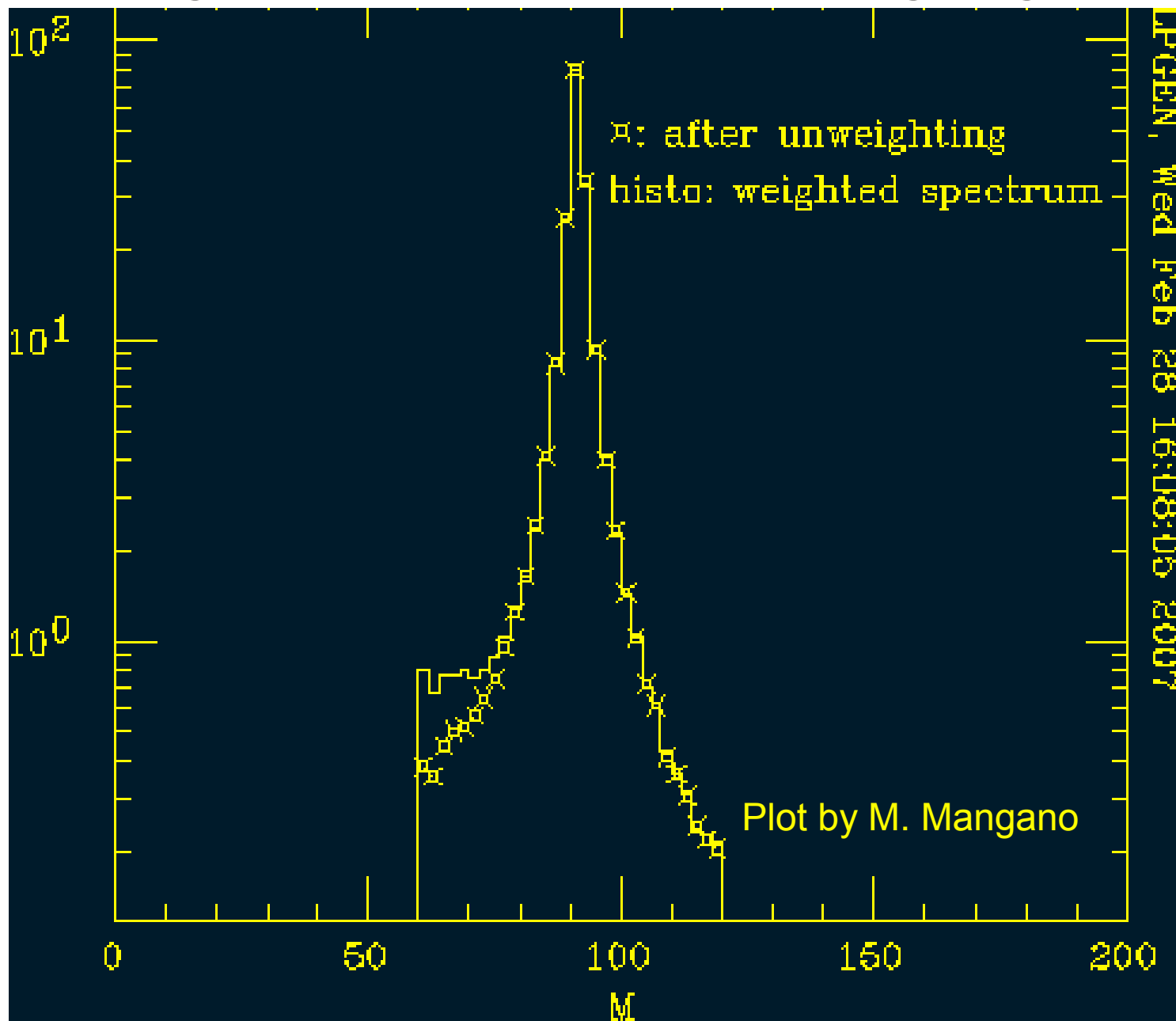
Di-muon Pt



Alpgen weirdness : $Z \rightarrow \mu\mu + 0$ jet exclusive matching



Alpgen weirdness : related to unweighting



Conclusion

Alpgen is in use in D0 but with corrections :

- K factors for the cross section.
- Vector boson Pt reweighting.
- Correct for known bug/problem with ad-hoc solution when a bug is known (until a corrected Alpgen version pops out)